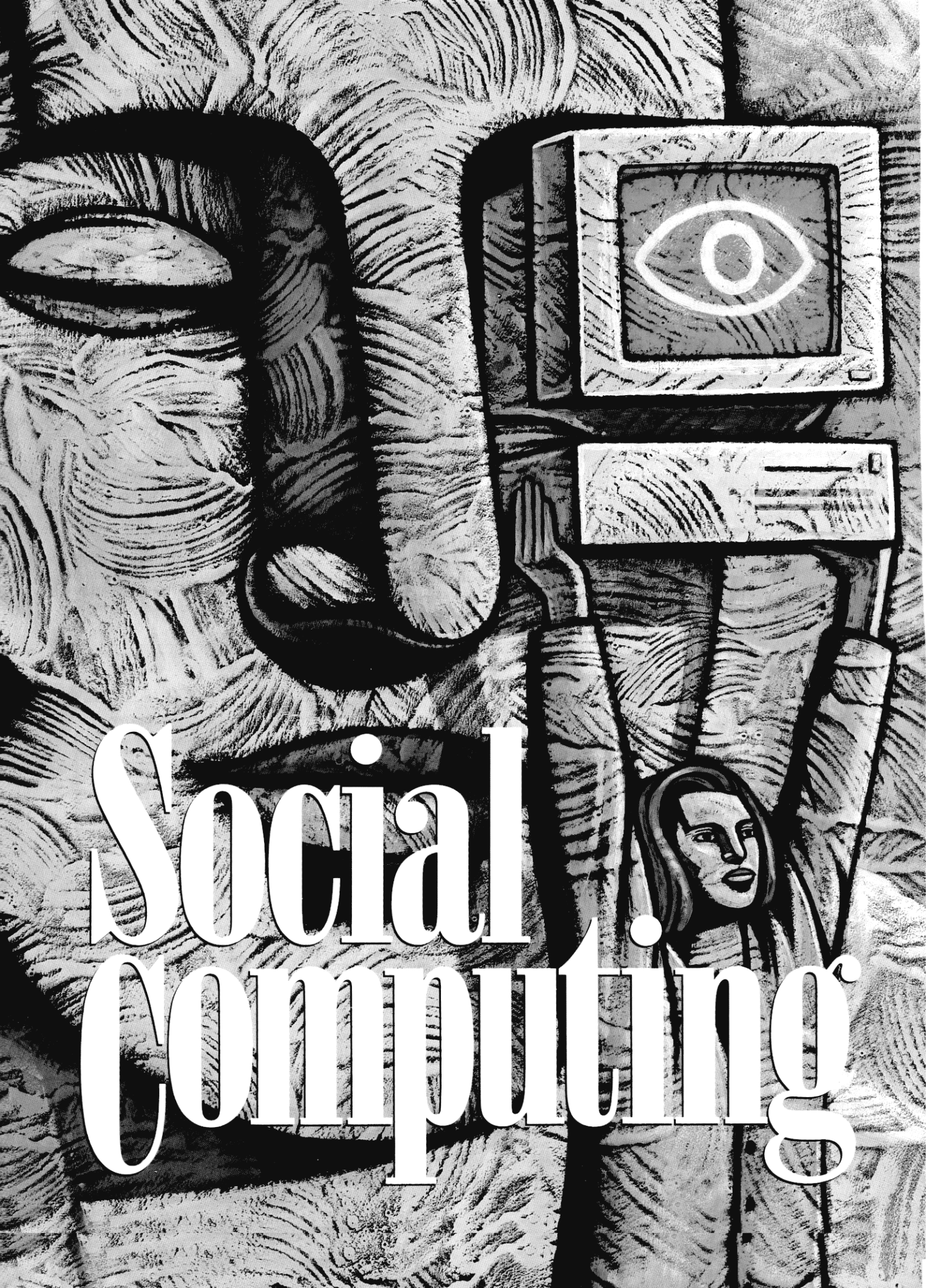
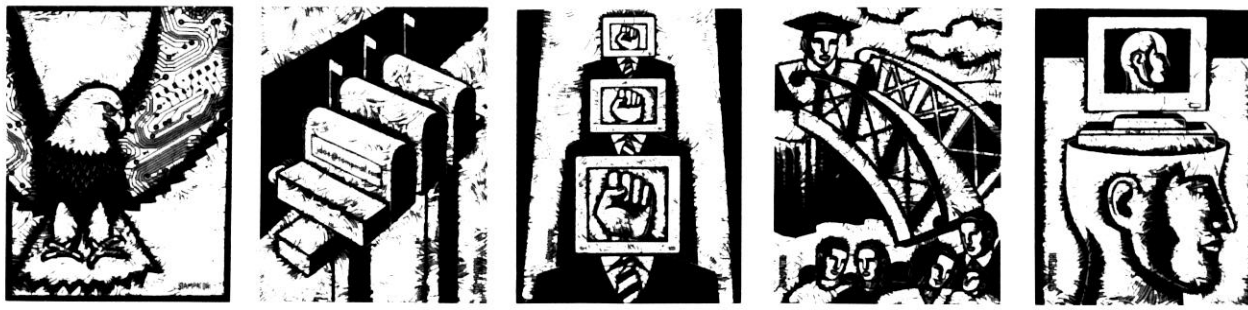

Social
Computing

January 1994
COMMUNICATIONS
OF THE ACM



DOUG SCHULER
Guest Editor

Computational power is frequently touted as the silver bullet that will tame many of our contemporary crises. But what basis in fact is there for these leaps of technological enthusiasm? Are they just convenient rationalizations for not perturbing the status quo? Computers *do* have the potential to alter our lives in new ways, but how?

What is the nature of *social computing* and what can we do about it?

If software were developed by individuals for their own use at their own behest, the social nature of computing might not exist. Of course, the opposite scenario is closer to the truth: groups of people develop software for others to use—usually at the behest of other people. Software gels intentions into a series of instructions that can be transferred from one location to another.

Social computing describes any type of computing application in which software serves as an intermediary or a focus for a social relation:

- when people communicate via newsgroups such as email or multiuser dungeons (MUDs);
- when governments devise policies involving software and network development;
- when people learn and teach about computers;
- when groups of people develop software;
- when private records are kept on individuals;
- when workplace tasks are defined by software; and
- when decisions of life and death are influenced by software.

I would like to say this special section covers all the areas circumscribed by the concept of social computing. Although we fall short of that unattainable goal, I am pleased at the deep and diverse forays our featured authors have made into this murky territory. These articles explore some of the more important perspectives of social computing and illustrate some of the thorny issues we have before us. Each author highlights critical questions that must be addressed:

How have international relationships shaped the development of computing and what opportunities now exist for defining new priorities? Gary Chapman discusses how the changed international climate presents a unique opportunity for us to redefine a national vision, and makes recommendations on how to accomplish it.

What are community members doing with computers to address community, educational, and democratic needs at the local level? Doug Schuler details the community computing movement and its principles, opportunities, and risks.

How do computers in the workplace act to maintain or alter power relations among employees? Andrew Clement explains how power relationships are defined and maintained through software mediation. His article is a provocative and perceptive study of the notion of computers and empowerment.

How are the social and technical spheres linked and how are these links accommodated? Batya Friedman and Peter H. Kahn discuss educational approaches that will help developers of technological systems better integrate social needs.

If software causes harm, who is accountable? Helen Nissenbaum's article examines the problem of a computerized society devoid of accountability, and addresses the challenge of maintaining accountability for the impacts of computing

The social nature of software is inescapable. In each example featured here, software established social relations at a distance, modifying existing social patterns. When this phenomenon repeats over and over again, the implications—both explicit and implicit—are profound. Moreover, these new relations are forming the initial players on which future relations will be built. The design decisions that are being made now will deeply influence the way we interact with others in the future.

Participating in the development of social computing is an opportunity we simply can not ignore.

Illustrations by Riccardo Stampatori, Three in A Box